

## AMENDMENTS TO THE CLAIMS

1. (canceled)

2. (canceled)

3. (canceled)

4. (currently amended) A method as recited in Claim 1, wherein:

the second lock data structure further comprises a reference number;

~~said step of creating a lock data structure further comprises setting the reference~~

~~number set to a predetermined initial value; and~~

~~said method further comprises, if it is determined to grant the request, then replacing~~

~~the value of the reference number in the lock data structure with a sum of the~~

~~value of the reference number in the lock data structure and a predetermined~~

~~reference change value~~

the method further comprising the steps of:

receiving the second lock to be released having data indicating the particular resource

object;

determining whether the reference number of the second lock to be released

substantially equals the predetermined initial value of the reference number;

and

if it is determined the reference number of the second lock to be released does not

substantially equal the predetermined initial value, then replacing the value of

the reference number in the second lock with a difference substantially equal

18 to the value of the reference number in the second lock minus a predetermined  
19 reference change value.

1 5. (canceled)

1 6. (currently amended) A method as recited in Claim 5 4, further comprising, if it is  
2 determined the reference number of the second lock to be released substantially  
3 equals the predetermined initial value, then deleting the second lock data structure for  
4 the particular resource object.

1 7. (canceled)

1 8. (currently amended) A method of updating a resource object using optimistic locks,  
2 the method comprising the computer-implemented steps of:  
3 receiving from a client process a request to update a particular resource object;  
4 sending to a lock manager process a request for a first lock for access to the particular  
5 resource object, the request including data indicating an optimistic lock type;  
6 receiving the first lock for access to the particular resource object, the first lock  
7 including data indicating the resource object, the optimistic lock type and a  
8 first value for a version number related to a number of changes to the resource  
9 object since the lock manager generated a lock data structure corresponding to  
10 the resource object; and  
11 updating the resource object by

cl

12 sending to a lock manager process a request for a second lock for access to the  
13 particular resource object, the request including data indicating the  
14 resource object identification and an exclusive lock type;  
15 receiving the second lock for access to the particular resource object, the  
16 second lock including data indicating the resource object  
17 identification, the exclusive lock type and a second value for the  
18 version number;  
19 determining whether the second value for the version number substantially  
20 equals the first value for the version number; and  
21 if the second value substantially equals the first value, then  
22 committing an updated resource object to the resource, and  
23 replacing the second value in for the reference version number in the  
24 second lock with a third value of for the version number, the  
25 third value computed by adding the second value and a  
26 predetermined version change value.

1 9. (original) The method as recited in Claim 8, further comprising, if the second value  
2 does not substantially equal the first value, then sending a message to the client  
3 process, the message indicating that the resource object was not updated.

1 10. (previously presented) The method as recited in Claim 8, further comprising sending  
2 to the lock manager process a first release message to release the first lock.

1 11. (previously presented) The method as recited in Claim 10, further comprising sending  
2 to the lock manager process a second release message to release the second lock.

cl

1 12. (previously presented) The method as recited in Claim 9, further comprising sending  
2 to the lock manager process a release message to release the second lock, the release  
3 message including data indicating the third value of the version number in the second  
4 lock and the exclusive lock type, wherein the third value of the version number is  
5 used by the lock manager to replace the second value of the version number in the  
6 lock data structure.

1 13. (canceled)

1 14. (currently amended) A computer-readable medium carrying one or more sequences of  
2 instructions for updating a resource object, which instructions, when executed by one  
3 or more processors, cause the one or more processors to carry out the steps of:  
4 receiving from a client process a request to update a particular resource object;  
5 sending to a lock manager process a request for a first lock for access to the particular  
6 resource object, the request including data indicating an optimistic lock type;  
7 receiving the first lock for access to the particular resource object, the first lock  
8 including data indicating the resource object, the optimistic lock type and a  
9 first value for a version number related to a number of changes to the resource  
10 object since the lock manager generated a lock data structure corresponding to  
11 the resource object; and  
12 updating the resource object by  
13 sending to a lock manager process a request for a second lock for access to the  
14 particular resource object, the request including data indicating the  
15 resource object identification and an exclusive lock type;

C1

16 receiving the second lock for access to the particular resource object, the  
17 second lock including data indicating the resource object  
18 identification, the exclusive lock type and a second value for the  
19 version number;  
20 determining whether the second value for the version number substantially  
21 equals the first value for the version number; and  
22 if the second value substantially equals the first value, then  
23 committing an updated resource object to the resource, and  
24 replacing the second value ~~in~~ for the ~~reference~~ version number in the  
25 second lock with a third value ~~of~~ for the version number, the  
26 third value computed by adding the second value and a  
27 predetermined version change value.

1 15. (canceled)

1 16. (currently amended) An apparatus for updating a resource object, comprising:  
2 a processor;  
3 one or more stored sequences of instructions which, when executed by the processor,  
4 cause the processor to carry out the steps of:  
5 receiving from a client process a request to update a particular resource object;  
6 sending to a lock manager process a request for a first lock for access to the  
7 particular resource object, the request including data indicating an  
8 optimistic lock type;

9 receiving the first lock for access to the particular resource object, the first  
10 lock including data indicating the resource object, the optimistic lock  
11 type and a first value for a version number related to a number of  
12 changes to the resource object since the lock manager generated a lock  
13 data structure corresponding to the resource object; and  
14 updating the resource object by  
15 sending to a lock manager process a request for a second lock for  
16 access to the particular resource object, the request including  
17 data indicating the resource object identification and an  
18 exclusive lock type;  
19 receiving the second lock for access to the particular resource object,  
20 the second lock including data indicating the resource object  
21 identification, the exclusive lock type and a second value for  
22 the version number;  
23 determining whether the second value for the version number  
24 substantially equals the first value for the version number; and  
25 if the second value substantially equals the first value, then  
26 committing an updated resource object to the resource, and  
27 replacing the second value ~~in~~ for the reference version number  
28 in the second lock with a third value ~~of~~ for the version  
29 number, the third value computed by adding the second  
30 value and a predetermined version change value.

1 17. (canceled)

C/

1 18. (currently amended) An apparatus for updating a resource object, comprising:  
2 a means for receiving from a client process a request to update a particular resource  
3 object;  
4 a means for sending to a lock manager process a request for a first lock for access to  
5 the particular resource object, the request including data indicating an  
6 optimistic lock type;  
7 a means for receiving the first lock for access to the particular resource object, the  
8 first lock including data indicating the resource object, the optimistic lock type  
9 and a first value for a version number related to a number of changes to the  
10 resource object since the lock manager generated a lock data structure  
11 corresponding to the resource object; and  
12 a means for updating the resource object, including  
13 a means for sending to a lock manager process a request for a second lock for  
14 access to the particular resource object, the request including data  
15 indicating the resource object identification and an exclusive lock type;  
16 a means for receiving the second lock for access to the particular resource  
17 object, the second lock including data indicating the resource object  
18 identification, the exclusive lock type and a second value for the  
19 version number;  
20 a means for determining whether the second value for the version number  
21 substantially equals the first value for the version number;  
22 a means for committing an updated resource object to the resource if the  
23 second value substantially equals the first value; and

C

24 a means for replacing the second value ~~in~~ for the ~~reference~~ version number in  
25 the second lock with a third value of the version number if the second  
26 value substantially equals the first value, the third value computed by  
27 adding the second value and a predetermined version change value.

1 19. (previously presented) The computer-readable medium as recited in Claim 14,  
2 wherein the instructions, when executed by one or more processors, cause the one or  
3 more processors to carry out the step of:  
4 if the second value does not substantially equal the first value, then sending a message  
5 to the client process, the message indicating that the resource object was not  
6 updated.

1 20. (previously presented) The computer-readable medium as recited in Claim 14,  
2 wherein the instructions, when executed by one or more processors, cause the one or  
3 more processors to carry out the step of:  
4 sending to the lock manager process a first release message to release the first lock.

1 21. (previously presented) The computer-readable medium as recited in Claim 20,  
2 wherein the instructions, when executed by one or more processors, cause the one or  
3 more processors to carry out the step of:  
4 sending to the lock manager process a second release message to release the second  
5 lock.

22. (previously presented) The computer-readable medium as recited in Claim 19,  
wherein the instructions, when executed by one or more processors, cause the one or  
more processors to carry out the step of:  
sending to the lock manager process a release message to release the second lock, the  
release message including data indicating the third value of the version  
number in the second lock and the exclusive lock type, wherein the third value  
of the version number is used by the lock manager to replace the second value  
of the version number in the lock data structure.

23. (previously presented) The apparatus as recited in Claim 16, wherein the instructions,  
when executed by one or more processors, cause the one or more processors to carry  
out the step of:  
if the second value does not substantially equal the first value, then sending a message  
to the client process, the message indicating that the resource object was not  
updated.

24. (previously presented) The apparatus as recited in Claim 16, wherein the instructions,  
when executed by one or more processors, cause the one or more processors to carry  
out the step of:  
sending to the lock manager process a first release message to release the first lock.

25. (previously presented) The apparatus as recited in Claim 24, wherein the instructions,  
when executed by one or more processors, cause the one or more processors to carry  
out the step of:

4 sending to the lock manager process a second release message to release the second  
5 lock.

1 26. (previously presented) The apparatus as recited in Claim 23, wherein the instructions,  
2 when executed by one or more processors, cause the one or more processors to carry  
3 out the step of:

4 sending to the lock manager process a release message to release the second lock, the  
5 release message including data indicating the third value of the version  
6 number in the second lock and the exclusive lock type, wherein the third value  
7 of the version number is used by the lock manager to replace the second value  
8 of the version number in the lock data structure.

1 27. (previously presented) The apparatus as recited in Claim 18, further comprising:  
2 means for sending a message to the client process if the second value does not  
3 substantially equal the first value, the message indicating that the resource  
4 object was not updated.

1 28. (previously presented) The apparatus as recited in Claim 18, further comprising:  
2 means for sending to the lock manager process a first release message to release the  
3 first lock.

1 29. (previously presented) The apparatus as recited in Claim 28, further comprising:  
2 means for sending to the lock manager process a second release message to release  
3 the second lock.

1 30. (previously presented) The apparatus as recited in Claim 27, further comprising:

2 means for sending to the lock manager process a second release message to release the  
3 second lock, the second release message including data indicating the third value  
4 of the version number in the second lock and the exclusive lock type, wherein  
5 the third value of the version number is used by the lock manager to replace the  
6 second value of the version number in the lock data structure.

---

C2 1 31. (new) The computer-readable medium of Claim 14, wherein the second lock further  
2 comprises a reference number set to a predetermined initial value, and wherein the  
3 instructions cause the one or more processors to perform the steps of:  
4 receiving the second lock to be released having data indicating the particular resource  
5 object;  
6 determining whether the reference number of the second lock to be released  
7 substantially equals the predetermined initial value of the reference number;  
8 and  
9 if it is determined the reference number of the second lock to be released does not  
10 substantially equal the predetermined initial value, then replacing the value of  
11 the reference number in the second lock with a difference substantially equal  
12 to the value of the reference number in the second lock minus a predetermined  
13 reference change value.

1 32. (new) A computer-readable medium of Claim 31, wherein the instructions cause the  
2 one or more processors to perform the steps of:  
3 receiving the second lock to be released having data indicating the particular resource  
4 object;

5 determining whether the reference number of the second lock to be released  
6 substantially equals the predetermined initial value of the reference number;  
7 and  
8 if it is determined the reference number of the second lock to be released does not  
9 substantially equal the predetermined initial value, then replacing the value of  
10 the reference number in the second lock with a difference substantially equal  
11 to the value of the reference number in the second lock minus a predetermined  
12 reference change value.

1 33. (new) A computer-readable medium of Claim 32, wherein the instructions cause the  
2 one or more processors to perform the step of, if it is determined the reference number  
3 of the second lock to be released substantially equals the predetermined initial value,  
4 then deleting the second lock for the particular resource object.

1 34. (new) The apparatus of Claim 16, wherein the second lock further comprises a reference  
2 number set to a predetermined initial value, and wherein the instructions cause the one  
3 or more processors to perform the steps of:  
4 receiving the second lock to be released having data indicating the particular resource  
5 object;  
6 determining whether the reference number of the second lock to be released  
7 substantially equals the predetermined initial value of the reference number; and  
8 if it is determined the reference number of the second lock to be released does not  
9 substantially equal the predetermined initial value, then replacing the value of  
10 the reference number in the second lock with a difference substantially equal to

11 the value of the reference number in the second lock minus a predetermined  
12 reference change value.

C2 1 35. (new) The apparatus of Claim 34, wherein the instructions cause the one or more  
2 processors to perform the steps of:  
3 if it is determined the reference number of the second lock to be released substantially  
4 equals the predetermined initial value, then deleting the second lock for the  
5 particular resource object.

1 36. (new) The apparatus of Claim 18, wherein the second lock further comprises a reference  
2 number set to a predetermined initial value, the apparatus further comprising:  
3 means for receiving the second lock to be released having data indicating the particular  
4 resource object;  
5 means for determining whether the reference number of the second lock to be released  
6 substantially equals the predetermined initial value of the reference number; and  
7 means for replacing the value of the reference number in the second lock with a  
8 difference substantially equal to the value of the reference number in the second  
9 lock minus a predetermined reference change value, if it is determined the  
10 reference number of the second lock to be released does not substantially equal  
11 the predetermined initial value.

1 37. (new) The apparatus of Claim 36, further comprising:  
2 means for deleting the second lock for the particular resource object if it is determined  
3 the reference number of the second lock to be released substantially equals the  
4 predetermined initial value.

38. (new) A method for managing access to a resource, the method comprising the computer-implemented steps of:

receiving a request for access to a particular resource to make an update to the resource;

generating a lock associated with the particular resource, wherein the lock comprises information that indicates a first value for a version number that is equal to a value for a version number associated with the particular resource and is related to whether the particular resource has been updated;

receiving a request to commit the update to the particular resource;

determining whether a current value for the version number associated with the particular resource is equal to the first value for the version number indicated in the lock; and

if the current value for the version number is equal to the first value for the version number, then converting the lock to a different type of lock and committing the update.

39. (new) The method of Claim 38, wherein the step of converting the lock comprises converting the lock to an exclusive lock.

40. (new) A computer-readable medium carrying one or more sequences of instructions for managing access to a resource, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

receiving a request for access to a particular resource to make an update to the resource;

generating a lock associated with the particular resource, wherein the lock comprises information that indicates a first value for a version number that is equal to a

7 value for a version number associated with the particular resource and is related  
8 to whether the particular resource has been updated;  
9 receiving a request to commit the update to the particular resource;  
10 determining whether a current value for the version number associated with the  
11 particular resource is equal to the first value for the version number indicated in  
12 the lock; and  
13 if the current value for the version number is equal to the first value for the version  
14 number, then converting the lock to a different type of lock and committing the  
15 update.

1 41. (new) The computer-readable medium of Claim 40, wherein the instructions cause the  
2 one or more processors to perform the step of converting the lock by converting the lock  
3 to an exclusive lock.

1 42. (new) An apparatus for managing access to a resource, comprising:  
2 a processor;  
3 one or more stored sequences of instructions which, when executed by the processor,  
4 cause the processor to carry out the steps of:  
5 receiving a request for access to a particular resource to make an update to the  
6 resource;  
7 generating a lock associated with the particular resource, wherein the lock  
8 comprises information that indicates a first value for a version number  
9 that is equal to a value for a version number associated with the  
10 particular resource and is related to whether the particular resource has  
11 been updated;

12 receiving a request to commit the update to the particular resource;  
13 determining whether a current value for the version number associated with the  
14 particular resource is equal to the first value for the version number  
15 indicated in the lock; and  
16 if the current value for the version number is equal to the first value for the  
17 version number, then converting the lock to a different type of lock and  
18 committing the update.

1 43. (new) The apparatus of Claim 42, wherein the instructions cause the processor to  
2 perform the step of converting the lock by converting the lock to an exclusive lock.

1 44. (new) An apparatus for managing access to a resource, comprising:  
2 means for receiving a request for access to a particular resource to make an update to the  
3 resource;  
4 means for generating a lock associated with the particular resource, wherein the lock  
5 comprises information that indicates a first value for a version number that is  
6 equal to a value for a version number associated with the particular resource and  
7 is related to whether the particular resource has been updated;  
8 means for receiving a request to commit the update to the particular resource;  
9 means for determining whether a current value for the version number associated with  
10 the particular resource is equal to the first value for the version number indicated  
11 in the lock; and  
12 means for converting the lock to a different type of lock and committing the update if  
13 the current value for the version number is equal to the first value for the version  
14 number.